

--	--	--	--	--	--	--	--	--	--

MULTIMEDIA UNIVERSITY

FINAL EXAMINATION

TRIMESTER 1, 2019/2020

ECP1026 – ALGORITHMS AND DATA STRUCTURES (TE / RE)

25 OCTOBER 2019
3.00 p.m. – 5.00 p.m.
(2 Hours)

INSTRUCTIONS TO STUDENT

1. This Question paper consists of 6 pages including cover page with 4 Questions only.
2. Attempt **ALL** questions. All questions carry equal marks and the distribution of marks for each question is given.
3. Please write all your answers in the Answer Booklet provided.

Question 1

- (a) The electricity tariff rate for domestic usage is given in Table Q1.1

Table Q1.1

Domestic tariff (per month)	Pricing rate (RM/kWh)
First 200 kWh	0.218
201 – 300 kWh	0.334
301 – 600 kWh	0.516
601 kWh onwards	0.571
The minimum monthly charge is RM3.00	

Write a function named **calBill** that accepts a real-valued number indicating the electricity usage (in kWh) and a pointer to a real-valued number as function arguments. The function calculates the electricity charge (in RM) and places it on the pointer argument.

Demonstrate the correct use of the function **calBill** in a complete C program. The program inputs a real value from user, and passes it to the function for electricity charge calculation. Finally, the program displays the bill on the screen.

A sample program output is given in Figure Q1.1. (Texts in bold are user input)

Enter usage (kWh): 350
Electricity charge: RM102.80

Figure Q1.1

[15 marks]

- (b) Table Q1.2 shows the student enrollment data in the Electronics (EE) and Mechanical (ME) Engineering courses of a local university in the recent four years.

Table Q1.2

Academic year	Number of EE students	Number of ME students
2015	158	208
2016	121	162
2017	94	125
2018	85	102

Write C statements that perform each of the following tasks:

- Define a suitable record type for yearly student enrollment. [2 marks]
- Declare and initialize an array of record with the above data. [3 marks]
- Calculate the total number of enrollment. [5 marks]

Continued ...

Question 2

- (a) Figure Q2.1 shows an incomplete C program that input 5 integers from user in a linked list. The program displays the contents of the list and summation of all items. Based on the comments given, complete the program.

```
#include <stdio.h>
#include <stdlib.h>

struct Node {
    int key;
    struct Node *next;
};

int main() {
    int j=0, val, sum=0;
    struct Node *head;
    struct Node *curr;

    head=(struct Node *)malloc(sizeof(struct Node));
    head->key=0;
    head->next=NULL;
    curr=head;

    //Prompt user for 5 integers
    //Using a loop, repeat the following for 5 times
    //    Input an integer from user and store it in val
    //    Create a new node and assign the starting address of
    //        the allocated memory to curr->next
    //    Set the key member of the new node to val
    //    Set the next member of the new node to NULL
    //    Set curr to point at the new node

    while(head) {
        //Add the content of key member of the node pointed by head
        //    to sum
        //Print the content of key member of the node pointed by head

        head=head->next;
    }
    printf("\nSum = %d", sum);
    return 0;
}
```

Figure Q2.1

A sample program output is given in Figure Q2.2 (texts in **bold** are user inputs):

```
Enter 5 integers: 1 2 3 4 5
List contains 0 -> 1 -> 2 -> 3 -> 4 -> 5 ->
Sum = 15
```

Figure Q2.2

[12 marks]

Continued...

Question 2 (cont.)

(b) Consider the weighted graph given in Figure Q2.3.

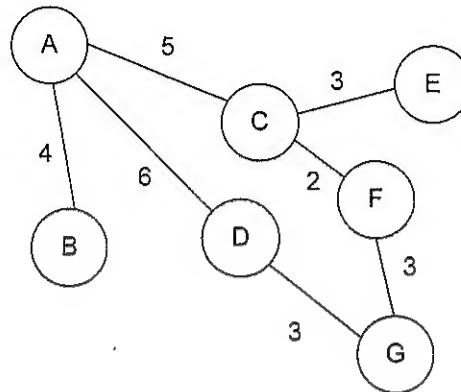


Figure Q2.3

- (i) Write a simple path that includes all vertices. [2 marks]
- (ii) Draw a spanning tree for the above graph. [2 marks]
- (iii) Represent the above graph in adjacency matrix form. Use -1 to indicate no direct path from one node to another. [3 marks]
- (iv) Write C declaration code that creates an array implementation of this graph. Use -1 to indicate no direct path from one node to another. [6 marks]

Question 3

(a) Consider the following mathematical formula:

$$f(n1, n2) = \begin{cases} n1, & \text{if } n2 = n1 \\ n2 + f(n1, n2 - 1), & \text{if } n2 > n1 \end{cases} \quad \text{where } n1 \leq n2.$$

- (i) Evaluate $f(1,4)$. [2 marks]
- (ii) What does the function $f(n1, n2)$ do? [2 marks]
- (iii) Write a C function to implement $f(n1, n2)$ using recursion. The function prototype is given as: `int f(int n1, int n2);` [6 marks]

Continued ...

Question 3 (cont.)

- (b) Figure Q3.1 shows a game tree in a game of Othello where a computer is playing white (W) and a human player is playing black (B). W1, W2, W3, B1, B2, B3, B4, B5, B6 and B7 represent the board positions, and the values below the nodes represent the total number of black pieces on the board.

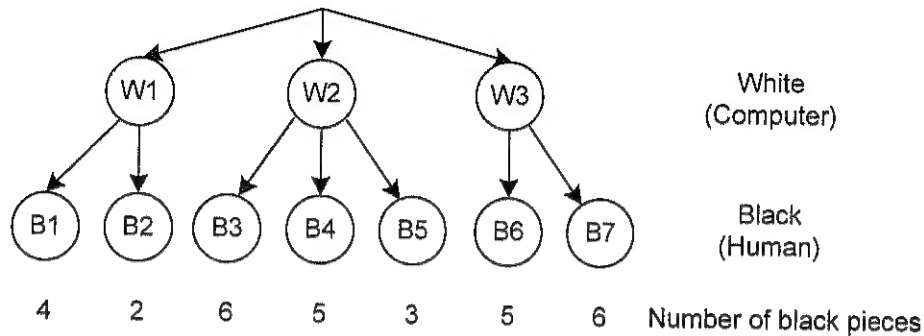


Figure Q3.1

The computer program uses a two-level look-ahead with minimax search approach. It is now the computer's turn. Based on the game tree, predict the computer's next move. Justify your answer. [3 marks]

- (c) Consider the program given in Figure Q3.2.

- (i) Indicate clearly the time complexity of each statement (if any). [8 marks]
- (ii) The probability of getting an odd value is the same as that of getting an even value, i.e., 0.5 each, determine the average-case time-complexity, worst-case time complexity and Big-O of the program. [2+1+1 marks]

```

1  #include<stdio.h>
2  #define N 200
3  int main(){
4      int k, n, a[N]={0}, cOdd=0, sumOdd=0, sumEven=0;
5
6      printf("Enter size of data: ");
7      scanf("%d", &n);
8
9      for(k=0; k<n; k++){
10         a[k] = k*k;
11
12         if(a[k]%2){
13             cOdd++;
14             sumOdd = sumOdd + a[k];
15         }
16         else
17             sumEven = sumEven + a[k];
18     }
19     printf("%d odd values with total = %d\n", cOdd, sumOdd);
20     printf("%d even values with total = %d\n", n-cOdd, sumEven);
21     return 0;
22 }
```

Figure Q3.2

Continued...

Question 4

- (a) Given the following items stored in an integer array:

22, 35, 30, 16, 8, 24, 15, 29, 46.

Search for a key value of **16** using both linear and binary search methods, respectively. Underline the item that is being compared with the key value **16** in each step.
[4+4 marks]

- (b) State what is hashing in the context of search, and describe how it is done.

[2+3 marks]

- (c) In arranging items in increasing order, selection sort works by selecting the smallest item from the unsorted portion of the array, and swapping it with the first of the unsorted portion of the array.

Based on the selection sort algorithm given in Figure Q4.1, write a function that implements selection sort on an integer array.

The function prototype is given as: **void selectionSort(int a[], int size);**

```
Set k1 to 0
while k1 is smaller than the size of array
  Set min to k1
  Set k2 to k1
  while k2 is smaller than the size of array
    if a[k2] is smaller than a[min]
      Set min to k2
    increment k2 by 1

  Set temp to a[min]
  Set a[min] to a[k1]
  Set a[k1] to temp
  increment k1 by 1
```

Figure Q4.1

[12 marks]

End of Paper